

hadoop 安装手册

<V1.0>

编写：朱自林 魏伦凯 周园园

审核：肖冰 王兵 李艳红

批准：高奎

©北京锐安科技有限公司

北京海淀区中关村南大街乙 56 号方圆大厦 9 层 100044

电话：(010) 58719666

传真：(010) 58719666-9588

服务电话：(010) 58719666

版本号	变化状态	修改点说明	变更人	存放位置	审批人	审批日期
V0.1	C		朱自林			
V0.2	M	配置 namenode 为 HA 模式	周园园 魏伦凯			
V1.0	M	修 改 安 装 zookeeper 的描 述	朱自林			

变更记录

*变化状态：C——创建，A——增加，M——修改，D——删除

修改点说明：对变化状态进行简单解释，如增加了某项功能，修改了某个模块等信息。

目录

1	前言	5
1.1	文档目的	5
1.2	读者对象	5
1.3	参考文献	5
2	安装硬件环境要求	5
3	安装过程	6
3.1	创建用户组和用户	9
3.2	安装 JDK	7
3.3	修改主机名	8
3.4	配置 SSH 无密码登录	错误!未定义书签。
3.5	关闭防火墙	10
3.6	Hadoop 安装配置	17
3.6.1	安装 hadoop 程序	错误!未定义书签。
3.6.2	配置 Hadoop 集群	错误!未定义书签。
3.6.3	格式化 HDFS 文件系统	22
3.6.4	Hadoop 的使用	23
3.7	Zookeeper 安装配置	12
3.7.1	zookeeper 安装	错误!未定义书签。
3.7.2	修改配置文件	13
3.7.3	创建 myid 文件	14
3.7.4	启动服务	15
3.7.5	检查服务启动状态	15
3.8	hbase 安装配置	26
3.8.1	安装 hbase 程序	26
3.8.2	配置 hbase	26
3.8.3	hbase 的使用	28

1 前言

1.1 文档目的

本文描述了 NoSql 部分的安装步骤，包括 JDK 安装过程及配置、无密码 SSH 登录过程及配置、hadoop 集群的安装过程及配置、zookeeper 的安装过程及配置、hbase 的安装过程及配置。

注：如没有明确指出，则默认的操作用户为 hadoop

如没有明确指出，则命令后的 # 代表的含义为注释，请不要将#当做命令的一部分

1.2 读者对象

现场工程、运维、客服人员。

1.3 参考文献

2 安装硬件环境要求

CPU：6 核*2 以上

内存：16*4 DDR4

操作系统：CentOS6.4 64 位

网络条件：千兆网络环境

硬盘：1T*12 SATA2.0 7200Rmp 企业级硬盘

其他部分有关硬件要求的描述，以此处的内容为准。

3 机器分配

假设有 5 台机器，主机名称为 hadoop1、hadoop2、hadoop3、hadoop4、hadoop5，

4.4 章节中会进行配置。

说明：

1. NameNode、DataNode、JournalNode 属于 hadoop 进程；
2. HMaster 和 HRegionServer 属于 Hbase 进程；
3. ZK 为 Zookeeper，ZK 的配置只能配置为奇数台，3、5、7 等。（如果集群规模为 10 台以内，则配置 3 台；如果规模为 10-40 左右，则配置 5 台；如果超过 40 台，则配置 7 台；这只是一个参考值，并不需要严格按照这个标准，如：集群规模为 11，也可以只配置 3 台 ZK）

4. 当机器数量比较多时，可让 NameNode 和 HMaster 节点用专门的机器部署

	Hadoop 集群			ZK	Hbase 集群	
	NameNode	DataNode	JournalNode		HMaster	HRegionServer
hadoop1	✓	✓			✓	✓
hadoop2	✓	✓				✓
hadoop3		✓	✓	✓		✓
hadoop4		✓	✓	✓		✓
hadoop5		✓	✓	✓		✓

1. hadoop 所有机器均安装，其中 hadoop1 和 hadoop2 安装 NameNode 作为 Hadoop 的主节点，并且需要在安装 zk 的节点上面启动 JournalNode 进程；
2. Hbase 所有机器均安装，其中 hadoop1 作为 Hbase 的主节点 HMaster；
3. ZK 为 zookeeper 缩写，zk 只能安装奇数个节点，本例中只安装 hadoop3、hadoop4、hadoop5 三台机器；

4 安装过程

4.1 准备过程

在安装下面 rpm 包之前，先将 rpm 包放到 YUM 仓库服务器的指定目录下

需要使用到的 rpm 包有：

jdk-7u75-linux.x64.pm

hadoop-sshkey-1.0.0-00001.x86_64.rpm

zookeeper-3.4.5-16.x86_64.rpm

hadoop-2.5.0-cdh5.2.1-1.x86_64.rpm

hbase-0.98.6-16.x86_64.rpm

oozie-4.0.0-00001.x86_64.rpm

rpm 包的具体存放路径和创建索引的命令请参考：《YUM 仓库安装和使用手册.doc》中的 3.1 节“增加发布软件包”

注：rpm 包的安装顺序参照文档中的章节顺序

4.2 目录结构：

软件通过 yum 方式安装，安装后的软件统一放在 /home/hadoop/yarn 目录下面。

4.3 安装 JDK

请以 **root** 用户身份执行下列命令和配置，这个步骤所有机器都必须安装。

Yum 方式安装 jdk：

JDK 版本：jdk-7u75-linux-x64

安装路径：/usr/java/jdk1.7.0_75

```
[root@nosqlmaster ~]# yum clean all
```

```
[root@nosqlmaster ~]# yum clean jdk
```

1. 配置环境变量，vi /etc/profile，在行末尾添加

```
export JAVA_HOME=/usr/java/jdk1.7.0_75
```

```
export CLASSPATH=.:$JAVA_HOME/lib/tools.jar:$JAVA_HOME/lib/dt.jar
```

```
export PATH=$JAVA_HOME/bin:$PATH
```

2. 执行命令：

```
[root@nosqlmaster ~]# source /etc/profile
```

使配置立即生效

3. 执行命令：java -version

```
[root@nosqlmaster ~]# java -version
```

查看是否安装成功

```
[root@nosqlmaster java]# java -version
java version "1.7.0_75"
Java(TM) SE Runtime Environment (build 1.7.0_75-b13)
Java HotSpot(TM) 64-Bit Server VM (build 24.75-b04, mixed mode)
```

若如上图显示，则表示安装成功

4.4 修改主机名

请以 **root** 用户身份执行下列命令和配置，这个步骤所有机器都必须安装。

- 1, 连接到主节点，修改 network，执行

```
[root@nosqlmaster ~]# vi /etc/sysconfig/network
```

修改 HOSTNAME=hadoop1

- 2, 执行命令：

```
[root@nosqlmaster ~]# hostname hadoop1
```

#hostname 后面的名称根据具体的主机名称进行修改，依次在集群中的每台

机器上执行 1-2 步骤。

- 3, 在命令行执行：exit 后重新连接，即可看到主机名修改成功与否

- 4, 在主节点上修改 hosts 文件，执行 vi /etc/hosts，在行末尾添加

```
168.5.15.112 hadoop1
```

```
168.5.15.116 hadoop2
```

```
168.5.15.207 hadoop3
```

```
168.5.15.201 hadoop4
```

```
168.5.15.202 hadoop5
```

注：IP 地址和主机名需要根据具体的机器 IP 地址和主机名进行修改

需要将 solr 集群的 IP 地址、主机名也添加到该文件中（solr 集群指的是信息检索组搭建的集群）

假设 solr 集群中的机器有 192.168.251.160、192.168.251.161、192.168.251.162 三台，hostname 分别为 runs251160、runs251161、runs251162，则在/etc/hosts 文件中再添加上下面三行

```
192.168.251.160 runs251160
```

```
192.168.251.161 runs251161
```

```
192.168.251.162 runs251162
```

- 5, 将主节点修改完的 hosts 文件使用 scp 命令（使用 **root** 用户执行 scp 命令）传输到集群中的所有机器：


```
[root@nosqlmaster ~]# scp /etc/hosts root@hadoop2:/etc/hosts
```

#依次替换集群上所有节点的 hosts 文件

4.5 关闭防火墙

请以 **root** 用户身份执行下列命令和配置，这个步骤所有机器都必须执行。

临时关闭

查看状态：

```
[root@nosqlmaster ~]# service iptables status
```

开启：

```
[root@nosqlmaster ~]# service iptables start
```

关闭：

```
[root@nosqlmaster ~]# service iptables stop
```

永久关闭

开启：

```
[root@nosqlmaster ~]# chkconfig iptables on
```

关闭：

```
[root@nosqlmaster ~]# chkconfig iptables off
```

查看状态时如果出现：

iptables: Firewall is not running.

说明防火墙关闭成功

4.6 Linux 环境配置

配置 hadoop 最大文件打开数：Centos6.4 (RHL6.X)

```
[root@nosqlmaster ~]# vi /etc/security/limits.d/90-nproc.conf
```

在文件结尾添加下面两行内容：

```
hadoop soft nfile unlimited
```

```
hadoop soft nproc unlimited
```

4.7 时间同步

请以 **root** 用户身份执行下列命令和配置，只需要在其中的一台上启动 **ntp server** 即可，假设启动 **ntp server** 服务的机器主机名为 **hadoop1**

4.7.1 编译安装 ntp server

```
rpm -qa | grep ntp
```

若没有找到，则说明没有安装 ntp 包，从光盘上找到 ntp 包，使用

```
rpm -Uvhntp***.rpm
```

进行安装

4.7.2 修改 ntp.conf 配置文件

下面两种方式选择一种即可，如果所有的机器都在一个网段内，则可以选择下面两种中的任意一种，如果不在同一个网段则只能选择第一种

```
[root@nosqlmaster ~]# vi /etc/ntp.conf
```

在文件中添加下面的两句话：

```
server 127.127.1.0  
fudge 127.127.1.0 stratum 8
```

方式一，将“restrict default kod nomodify notrap nopeer noquery”这行修改成：restrict default nomodify notrap（允许任何 IP 的客户机都可以进行时间同步）

方式二，在 restrict default nomodify notrap noquery（表示默认拒绝所有 IP 的时间同步）之后增加一行：

```
restrict 192.168.211.0 mask 255.255.255.0 nomodify notrap（只允许  
192.168.211.***网段的客户机进行时间同步）
```

4.7.3 启动 ntp 服务

在 hadoop1 机器上执行如下命令：

```
[root@nosqlmaster ~]# service ntpd start
```

设置为开机启动服务的命令为：

```
[root@nosqlmaster ~]# chkconfig ntpd on
```

启动完成后检查服务启动情况（如果显示为下图则表示启动成功）

```
[root@nosqlmaster ~]# service ntpd status
```

```
[root@hadoop1 hadoop]# service ntpd status
ntpd (pid 9873) is running...
```

注：ntpd 启动后，客户机要等几分钟再与其进行时间同步，否则会提示：“no server suitable for synchronization found”

4.7.4 执行同步命令

在集群中 hadoop2、hadoop3、hadoop4、hadoop5 上执行命令：

```
[root@nosqlmaster ~]# ntpdate hadoop1
```

如果显示下图的状态，则表示执行成功

```
[root@hadoop2 ~]# ntpdate hadoop1
8 Jun 15:34:01 ntpdate[27743]: adjust time server 192.168.251.12 offset 0.041833 sec
```

注：有可能需要多执行几次 ntpdate hadoop1 命令才能看到上图的效果

4.7.5 添加定时任务

在集群中 hadoop2、hadoop3、hadoop4、hadoop5 上执行命令：

```
[root@runs3035 /]# crontab -e
```

然后在打开的文本中添加下面一行内容，然后保存

```
0 21 * * * ntpdate hadoop1 >> /root/ntpdate.log 2>&1
```

hadoop1：将 hadoop1 修改成实际启动 ntp server 的机器主机名或者 IP 地址

注：上面的内容的含义为：每天 21 点定时执行下面的命令：

```
ntpdate hadoop1 >> /root/ntpdate.log 2>&1
```

4.8 添加机器集群互信

节点 **hadoop1、hadoop2、hadoop3**

下面以 **hadoop1** 节点为例(**hadoop2、hadoop3** 同样执行如下动作)

su - hadoop

ssh-keygen -t rsa (每个节点都需要执行生成公钥/密钥)

ssh-copy-id -i ~/.ssh/id_rsa.pub hadoop@hadoop1 (-p 2222)

ssh-copy-id -i ~/.ssh/id_rsa.pub hadoop@hadoop2 (-p 2222)

ssh-copy-id -i ~/.ssh/id_rsa.pub hadoop@hadoop3 (-p 2222)

hadoop2、hadoop3 为集群其他节点的 ip

-p 端口 默认端口不加**-p**，如果更改过端口，就得加上**-p**

注意：

sshd 为了安全，对属主的目录和文件权限有所要求。如果权限不对，则 ssh 的免密码登陆不生效。

用户目录权限为 755 或者 700，就是不能是 77x。

.ssh 目录权限一般为 755 或者 700。

rsa_id.pub 及 authorized_keys 权限一般为 644

rsa_id 权限必须为 600

登陆目标机器，查看 sshd 的日志信息。日志信息目录为，/var/log/secure

4.9 安装 Zookeeper 软件

请以 **root** 用户身份执行下列命令和配置，这个步骤只在 **hadoop3、hadoop4、hadoop5** 中执行。

yum 安装 zookeeper 软件：

zookeeper 版本：zookeeper-3.4.5-cdh5.2.1

Yum 安装：

```
[root@nosqlmaster ~]# yum clean all
```

```
[root@nosqlmaster ~]# yum install zookeeper
```

注：因为 zookeeper 依赖于 jdk，而 jdk 会自动导入环境变量，并且使环境变量生效。如果在安装 zookeeper 之前没有安装过 jdk，则需要关闭当前打开的连接远程桌面的窗口然后重新打开，环境变量才会生效；如果在安装 zookeeper 之前

jdk 已经安装完成并且环境变量已经配置正确并生效，则不需要关闭窗口。

验证：安装完 zookeeper 之后，用 hadoop 用户在安装 zookeeper 的机器之间进行 ssh 无密码登录，如果能够正常登录则安装正常，如果不能进行无密码登录则按照文档顺序继续执行下去，直到执行完（4.10 安装 hadoop 软件）之后再执行以下操作

分别检查 hadoop3、hadoop4 和 hadoop5 是否存在 /home/hadoop/.ssh/ 目录，如果目录不存在则创建目录（下面的几个操作使用 hadoop 用户）

```
[hadoop@hadoop4 ~]$ mkdir /home/hadoop/.ssh/
```

如果目录存在则删除 /home/hadoop/.ssh/ 下的文件

```
[hadoop@hadoop4 ~]$ rm -rf /home/hadoop/.ssh/*
```

然后从已经安装了 hadoop 但没有安装 zookeeper 的机器（如 hadoop1）上拷贝文件

```
[hadoop@hadoop4 ~]$ scp hadoop1:/home/hadoop/.ssh/* /home/hadoop/.ssh/
```

如果 zookeeper 已经启动，则重启 zookeeper

4.10 Zookeeper 配置

请以 **hadoop** 用户身份进行配置，这个步骤只在 **hadoop3**、**hadoop4**、**hadoop5** 中执行。

4.10.1 修改配置文件

```
[hadoop@nosqlmaster ~]# vi /home/hadoop/zookeeper/conf/zoo.cfg
```

修改配置文件中的内容

```
# The number of milliseconds of each tick
tickTime=5000
# The number of ticks that the initial
# synchronization phase can take
initLimit=10
# The number of ticks that can pass between
# sending a request and getting an acknowledgement
syncLimit=5
# the directory where the snapshot is stored.
```

```
# do not use /tmp for storage, /tmp here is just
# example sakes.
dataDir=/data1/zookeeper/data
dataLogDir=/data1/zookeeper/datalog

# the port at which the clients will connect
clientPort=2181
# the maximum number of client connections.
# increase this if you need to handle more clients
#maxClientCnxns=60
#
# Be sure to read the maintenance section of the
# administrator guide before turning on autopurge.
#
# http://zookeeper.apache.org/doc/current/zookeeperAdmin.html#sc_maintenance
#
# The number of snapshots to retain in dataDir
#autopurge.snapRetainCount=3
# Purge task interval in hours
# Set to "0" to disable auto purge feature
#autopurge.purgeInterval=1
server.0=hadop3:2888:3888
server.1=hadop4:2888:3888
server.2=hadop5:2888:3888
```

标红部分根据实际需要进行修改（除了 `server.0=hadop3:2888:3888` 这部分，其他内容可以使用默认值）

`dataDir=/data1/zookeeper/data` : 为 zookeeper 数据存放位置，如果目录不存在则创建
`dataLogDir=/data1/zookeeper/datalog` : 为 zookeeper 日志数据存放位置，如果目录不存在则创建
`server.x= yyyy:2888:3888` : 将 yyyy 替换成 zookeeper 节点的主机名，如果超过三台，请自行添加，x 值递增

4.10.2 创建 myid 文件

在上一步骤中配置文件中配置的数据存放路径为 `dataDir=/data1/zookeeper/data`，在 `/data1/zookeeper/data` 下创建名为 myid 的文件

根据 zoo.cfg 配置文件中的下列内容修改 myid 文件

```
server.0=hadop3:2888:3888
server.1=hadop4:2888:3888
server.2=hadop5:2888:3888
```

如：

`server.0=hadop3:2888:3888`，则将 hadop3 的 myid 内容修改为：0（myid 文件中只有一个 0，没有其他内容）

`server.1=hadop4:2888:3888`，则将 hadop4 的 myid 内容修改为：1（myid 文件中只有一个 1，没有其他内容）

以此类推，hadoop5 的 myid 内容为 2

4.10.3 启动服务

在每台机器的/home/hadoop/zookeeper/bin 目录下
执行命令：`./zkServer.sh start`

4.10.4 关闭服务

在每台机器的/home/hadoop/zookeeper/bin 目录下
执行命令：`./zkServer.sh stop`

关闭 zookeeper 之前需要先关闭 hbase 和 hadoop
正确的启动顺序为：zookeeper—>hadoop—>hbase
正确的关闭顺序为：hbase—>hadoop—>zookeeper

4.10.5 检查服务启动状态

在 hadoop3、hadoop4、hadoop5 三个节点中执行如下命令：
`sh /home/hadoop/zookeeper/bin/zkServer.sh status`

```
[hadoop@hadoop3 zookeeper]$ sh /home/hadoop/zookeeper/bin/zkServer.sh status
JMX enabled by default
Using config: /home/hadoop/zookeeper/bin/./conf/zoo.cfg
Mode: follower

[hadoop@hadoop4 ~]$ sh /home/hadoop/zookeeper/bin/zkServer.sh status
JMX enabled by default
Using config: /home/hadoop/zookeeper/bin/./conf/zoo.cfg
Mode: leader

[hadoop@hadoop5 ~]$ sh /home/hadoop/zookeeper/bin/zkServer.sh status
JMX enabled by default
Using config: /home/hadoop/zookeeper/bin/./conf/zoo.cfg
Mode: follower
```

如果显示上图的情况则表示 zookeeper 启动成功

注：

1. zookeeper 中其他的配置文件都一样，但是 myid 文件一定不能一样
2. 如果 zookeeper 启动之后报错或者启动失败，有可能是以下原因造成的
 - a) 请检查 myid 文件中的内容是否与 zoo.cfg 文件中配置的内容一致（具体请参见第 4.10.2 步骤的“创建 myid 文件”）
 - b) 防火墙是否关闭（具体请参见 4.5 节“关闭防火墙”）
 - c) JAVA_HOME 设置是否正确（具体请参见 4.3 节“安装 JDK”）
 - d) zoo.cfg 文件中设置的 `clientPort=2181` 端口号是否被占用

检查命令为：

`netstat -lntup` 或 `lsof -i:2181`

检查端口是否被占用，如果被占用，则修改 zoo.cfg 文件中的 `clientPort=2181`，将端口换成其他没有被占用的端口（所有 zookeeper 机器都要修改）

4.11 安装 hadoop 软件

请以 **root** 用户身份执行下列命令，这个步骤在所有机器中执行。

yum 安装 hadoop 软件：

```
[root@nosqlmaster ~]# yum clean all
[root@nosqlmaster ~]# yum install hadoop
```

注：因为 hadoop 依赖于 jdk，而 jdk 会自动导入环境变量，并且使环境变量生效。如果在安装 hadoop 之前没有安装过 jdk，则需要关闭当前打开的连接远程桌面的窗口然后重新打开，环境变量才会生效；如果在安装 hadoop 之前 jdk 已经安装完成并且环境变量已经配置正确并生效，则不需要关闭窗口

4.12 Hadoop 配置

Hadoop 是分布式集群系统，提供分布式文件系统 HDFS 和 MapReduce 编程框架；安装前需要设计好 Hadoop 集群（包括集群节点数量、master 所在节点、slave 节点数、各节点 IP 地址等），请参考 3 机器分配。

对每一个集群节点，建议进行全新安装（指全新安装 CentOS 6.4 Linux 操作系统和 Hadoop 集群），以避免原有系统残留信息造成的干扰，引起安装过程中的各种异常。

以下步骤需要在 **hadoop 集群** 中每一台节点上执行，若有差别，会强调说明。

请以 **hadoop** 用户身份执行下列命令和配置。

4.12.1 配置 Hadoop 集群

Hadoop 版本：hadoop-2.5.0-cdh5.2.1

4.12.1.1 术语缩写表

缩写、术语	解释
/dataX 目录	指在系统根目录下创建的/data1, /data2, /data3, /data4 这样的目录，具体数量根据服务器的硬盘数量来定
Hadoop 主节点	也称作 Master 节点，将运行 Hadoop 的 namenode 和 Hbase 的 hmaster 进程

Hadoop 从节点	也称作 Slave 节点，将运行 Hadoop 的 datanode

以下所有配置文件都在/home/hadoop/hadoop/etc/hadoop/下

4.12.1.2 配置 core-site.xml 文件

按照以下说明进行配置（只修改标红的地方，请勿将下面的内容直接粘贴到服务器上的文件中）：

```
<?xml version="1.0"?>
<?xml-stylesheet type="text/xsl" href="configuration.xml"?>
<configuration>
<!--Hadoop configs -->
<property>
<name>fs.defaultFS</name>
<value>hdfs://runcdhInt</value>
</property>
<property>
<name>ha.zookeeper.quorum</name>
<value>hadoop3:2181,hadoop4:2181,hadoop5:2181</value>
</property>

<property>
<name>hadoop.tmp.dir</name>
<value>/data1</value>
</property>

<property>
<name>fs.checkpoint.period</name>
<value>3600</value>
</property>
<property>
<name>io.file.buffer.size</name>
<value>16384</value>
</property>
</configuration>
```

ha.zookeeper.quorum: 值修改成安装 zookeeper 的机器 IP 或主机名（端口默认采用 2181）

4.12.1.3 配置 hdfs-site.xml 文件（配置为高可用模式）

按照以下说明进行配置（只修改标红的地方，请勿将下面的内容直接粘贴到服务器上的文件中）：

```
<?xml version="1.0"?>
<?xml-stylesheet type="text/xsl" href="configuration.xml"?>
<configuration>
<property>
<name>dfs.nameservices</name>
<value>runcdhInt</value>
</property>
<property>
<name>dfs.ha.namenodes.runcdhInt</name>
<value>nn1,nn2</value>
</property>
<property>
<name>dfs.namenode.rpc-address.runcdhInt.nn1</name>
<value>hadoop1:8020</value>
</property>
<property>
<name>dfs.namenode.rpc-address.runcdhInt.nn2</name>
<value>hadoop2:8020</value>
</property>
```

```

</property>
<property>
  <name>dfs.namenode.http-address.runcdhInt.nn1</name>
  <value>hadoop1:50070</value>
</property>
<property>
  <name>dfs.namenode.http-address.runcdhInt.nn2</name>
  <value>hadoop2:50070</value>
</property>
<property>
  <name>dfs.namenode.shared.edits.dir</name>
  <value>qjournal://hadoop3:8485;hadoop4:8485;hadoop5:8485/runcdhInt</value>
</property>
<property>
  <name>dfs.client.failover.proxy.provider.runcdhInt</name>
  <value>org.apache.hadoop.hdfs.server.namenode.ha.ConfiguredFailoverProxyProvider</value>
</property>
<property>
  <name>dfs.journalnode.edits.dir</name>
  <value>/data1/dfs/jn</value>
</property>
<property>
  <name>dfs.ha.fencing.methods</name>
  <value>sshfence</value>
</property>
<property>
  <name>dfs.ha.fencing.ssh.private-key-files</name>
  <value>/home/hadoop/.ssh/id_rsa</value>
</property>
<property>
  <name>dfs.ha.fencing.ssh.connect-timeout</name>
  <value>30000</value>
</property>
<property>
  <name>dfs.ha.automatic-failover.enabled</name>
  <value>true</value>
</property>

<!-- Base configs -->
<property>
  <name>dfs.replication</name>
  <value>3</value>
</property>
<property>
  <name>dfs.namenode.name.dir</name>
  <value>file:///data1/dfs/name</value>
  <description>the data in every dir is same</description>
</property>

<property>
  <name>dfs.datanode.data.dir</name>
  <value>/data1/dfs/data,/data2/dfs/data,/data3/dfs/data,/data4/dfs/data,/data5/dfs/data,/data6/dfs/data,
/data7/dfs/data,/data8/dfs/data,/data9/dfs/data,/data10/dfs/data,/data11/dfs/data,/data12/dfs/data</value>
  <description>the data in every dir is not same</description>
</property>
<property>
  <name>dfs.webhdfs.enabled</name>
  <value>true</value>
</property>
<property>
  <name>dfs.permissions.enabled</name>
  <value>>false</value>
</property>

<property>
  <name>dfs.datanode.max.xcievers</name>
  <value>16000</value>
</property>
<property>
  <name>dfs.datanode.data.dir.perm</name>
  <value>755</value>
</property>
<property>
  <name>dfs.datanode.hdfs-blocks-metadata.enabled</name>
  <value>true</value>
</property>
</configuration>

```

dfs.namenode.rpc-address.runcdhInt.nn1: 修改值为 HA（高可用）的想要作为第一台 master 的 IP 地址

dfs.namenode.rpc-address.runcdhInt.nn2: 修改值为 HA（高可用）的想要作为第二台 master 的 IP 地址

dfs.namenode.http-address.runcdhInt.nn1: 修改值为 HA（高可用）的想要作为第一台 master

的 IP 地址

dfs.namenode.http-address.runcdhInt.nn2: 修改值为 HA（高可用）的想要作为第二台 master 的 IP 地址

dfs.namenode.shared.edits.dir: 配置 namenode 元数据共享存储位置，选择集群中任意 3 台主机作为 qjournal 提供服务

dfs.datanode.data.dir: 存储 hdfs 数据的目录，根据实际挂载盘数配置具体路径内容（现在默认要求挂载 12 块盘，如果确实挂载 12 块盘，则不需要修改此处内容；如果挂载盘数不为 12，则根据实际挂载盘数进行修改；假如有三块盘，则配置内容为

/data1/dfs/data,/data2/dfs/data,/data3/dfs/data)

4.12.1.4 配置 mapred-site.xml 文件（配置为 Yarn 模式）

按照以下说明进行配置（只需修改标红的地方，请勿将下面的内容直接粘贴到服务器上的文件中）：

```
<?xml version="1.0"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
<configuration>
  <property>
    <name>mapreduce.framework.name</name>
    <value>yarn</value>
  </property>
  <property>
    <name>mapreduce.jobhistory.address</name>
    <value>hadoop2:10020</value>
  </property>
  <property>
    <name>mapreduce.jobhistory.webapp.address</name>
    <value>hadoop2:19888</value>
  </property>
  <property>
    <name>mapreduce.map.memory.mb</name>
    <value>512</value>
  </property>
  <property>
    <name>mapreduce.reduce.memory.mb</name>
    <value>1024</value>
  </property>
  <property>
    <name>yarn.app.mapreduce.am.resource.mb</name>
    <value>1024</value>
    <description>mapreduce AppMaster memory</description>
  </property>
  <property>
    <name>mapreduce.application.classpath</name><value>$HADOOP_MAPRED_HOME/share/hadoop/mapreduce/*,$HADOOP_MAPRED_HOME/share/hadoop/mapreduce/lib/*,/home/hadoop/hbase/conf,/home/hadoop/hbase/lib/*</value>
  </property>
</configuration>
```

mapreduce.jobhistory.address: 配置 jobhistory 所在的机器 IP 地址（意思是：你设置 IP 地址是多少，jobhistory 就在哪台机器启动）

mapreduce.jobhistory.webapp.address: IP 地址同 mapreduce.jobhistory.address

4.12.1.5 配置 yarn-site.xml 文件（配置为高可用模式）

按照以下说明进行配置（只需修改标红的地方，请勿将下面的内容直接粘贴到服务器上的文件中）：

```
<?xml version="1.0"?>
<configuration>
<!--RM1 configs-->
<property>
```

```
<name>yarn.resourcemanager.address.rm1</name>
<value>hadoop1:23140</value>
</property>
<property>
  <name>yarn.resourcemanager.scheduler.address.rm1</name>
  <value>hadoop1:23130</value>
</property>
<property>
  <name>yarn.resourcemanager.webapp.https.address.rm1</name>
  <value>hadoop1:23189</value>
</property>
<property>
  <name>yarn.resourcemanager.webapp.address.rm1</name>
  <value>hadoop1:23188</value>
</property>
<property>
  <name>yarn.resourcemanager.resource-tracker.address.rm1</name>
  <value>hadoop1:23125</value>
</property>
<property>
  <name>yarn.resourcemanager.admin.address.rm1</name>
  <value>hadoop1:23141</value>
</property> >
<!-- RM2 configs -->
<property>
  <name>yarn.resourcemanager.address.rm2</name>
  <value>hadoop2:23140</value>
</property>
<property>
  <name>yarn.resourcemanager.scheduler.address.rm2</name>
  <value>hadoop2:23130</value>
</property>
<property>
  <name>yarn.resourcemanager.webapp.https.address.rm2</name>
  <value>hadoop2:23189</value>
</property>
<property>
  <name>yarn.resourcemanager.webapp.address.rm2</name>
  <value>hadoop2:23188</value>
</property>
<property>
  <name>yarn.resourcemanager.resource-tracker.address.rm2</name>
  <value>hadoop2:23125</value>
</property>
<property>
  <name>yarn.resourcemanager.admin.address.rm2</name>
  <value>hadoop2:23141</value>
</property>
</configuration>
```

上面的 yarn-site.xml 所有需要配置 hadoop1 和 hadoop2 的地方可以按照 hdfs-site.xml 文件中的 dfs.namenode.rpc-address.runcdhInt.nn1 和 dfs.namenode.rpc-address.runcdhInt.nn2 对应的 IP 设置

4.12.1.6 配置 slaves 文件

请清空此文件中的内容，然后将 Hadoop 集群所有 Slave 服务器的主机名称写入到此文件中，一行一个，如下所示：

hadoop1

hadoop2

hadoop3

hadoop4

hadoop5

注意 slaves 文件需要是 unix 格式，否则会出异常错误。

4.12.2 格式化 HDFS 文件系统

请以 **root** 用户身份执行下列命令和配置。

1. 机器初次安装无需执行如下 rm 操作

请在每台 Hadoop 节点服务器上，删除所有/dataX 目录下的文件

```
rm -rf /data?/dfs/*
```

注意：请慎重执行 rm 操作，以防数据丢失

2. 使用以下命令为/dataX 目录授权：

```
chown -R hadoop:hadoop /data?/
```

请以 **hadoop** 用户身份执行下列命令和配置。

1. 在 hadoop3、hadoop4、hadoop5（这些节点是在上面 hdfs.xml 文件中配置的 dfs.namenode.shared.edits.dir 对应的主机名称）节点分别上执行以下命令：

```
[hadoop@nosqlmaster ~]$ hadoop-daemon.sh start journalnode
```

注意：此命令启动 journalnode 进程

2. 在 Hadoop 的 hadoop1 节点上执行以下命令：

```
[hadoop@nosqlmaster ~]$ hadoop namenode -format
```

3. 根据 hdfs-site.xml 中 dfs.namenode.name.dir 的路径配置/data1, 将 hadoop1 中的/data1/dfs 复制到 hadoop2 的/data1 中：

```
[hadoop@hadoop2 ~]$ scp -r hadoop@hadoop1:/data1/dfs /data1
```

4. 在 hadoop1 节点上执行以下命令：

命令执行前保证 zookeeper 已启动

```
[hadoop@hadoop1 ~]$ hdfs zkfc -formatZK
```

以上步骤执行完毕后，HDFS 文件系统就初始化完成了。

4.12.3 Hadoop 的使用

4.12.3.1 启动 Hadoop

启动 hadoop 集群之前请先启动 zookeeper，如果 zookeeper 没启动请启动 zookeeper，待 zookeeper 启动成功后再启动 hadoop 集群。

请以 **hadoop** 用户身份执行下列命令和配置。

1. 登录 hadoop 集群的 hadoop1 节点上。
2. 执行

```
[hadoop@hadoop1~]$ sh /home/hadoop/hadoop/sbin/start-dfs.sh
```

命令启动 hdfs，启动后通过 jps 命令查看 master 节点 java 进程情况如下：

```
[hadoop@hadoop1 ~]$ jps
5644 DataNode
5539 NameNode
5746 DFSZKFailoverController
12664 Jps
```

其中 NameNode 是 hadoop 集群 master 节点命名节点服务程序，DataNode 是从节点服务程序。

通过 jps 命令查看任意 slave 节点 java 进程情况如下：

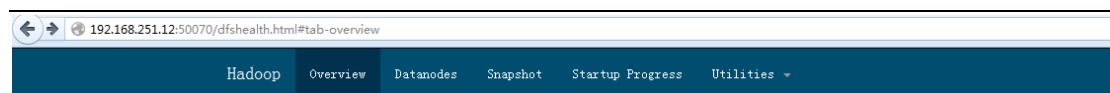
```
[hadoop@hadoop3 ~]$ jps
5644 DataNode
2762 Jps
```

其中 DataNode 是 hadoop 集群 slave 节点数据节点服务程序，主要查看 DataNode 进程是否启动。

可以通过浏览器访问：<http://hadoop1-IP:50070> 或 <http://hadoop2-IP:50070> 查看 hdfs 节点的健康状况

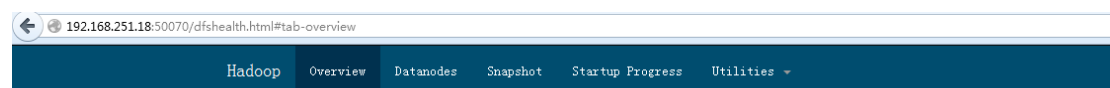
（上面的 hadoop1-IP 和 hadoop2-IP 为 hdfs-site.xml 文件中 `dfs.namenode.rpc-address.runcdhInt.nn1` 和 `dfs.namenode.rpc-address.runcdhInt.nn2` 属性对应的节点 IP 地址）

如果显示出下图的效果，一台状态为 active，一台为 standby，则表示启动成功



Overview 'hadoop1:8020' (active)

Started:	Tue Jun 09 17:48:04 CST 2015
Version:	2.5.0-cdh5.2.1, rd3c8b1c5c8dbba403570ae7c5fcd85d760534aa6
Compiled:	2014-11-19T19:01Z by jenkins from Unknown
Cluster ID:	CID-ff305870-5050-4a39-b809-08bae5256c33
Block Pool ID:	BP-1009261536-192.168.251.12-1433759377424



Overview 'hadoop2:8020' (standby)

Started:	Tue Jun 09 17:48:03 CST 2015
Version:	2.5.0-cdh5.2.1, rd3c8b1c5c8dbba403570ae7c5fcd85d760534aa6
Compiled:	2014-11-19T19:01Z by jenkins from Unknown
Cluster ID:	CID-ff305870-5050-4a39-b809-08bae5256c33
Block Pool ID:	BP-1009261536-192.168.251.12-1433759377424

3. 执行

```
[hadoop@hadoop1 ~]$ sh /home/hadoop/hadoop/sbin/start-yarn.sh
```

启动 yarn，启动后通过 jps 命令查看 master 节点 java 进程情况如下：

```
[hadoop@hadoop1 ~]$ jps
30787 NameNode
30927 DataNode
31363 ResourceManager
31203 DFSZKFailoverController
31472 NodeManager
20912 Jps
```

主要查看是否有 ResourceManager 和 Nodemanager 进程；

通过 jps 命令查看任意 slave 节点 java 进程情况如下：

```
[hadoop@hadoop3 ~]$ jps
16960 DataNode
32228 Jps
17157 NodeManager
16620 QuorumPeerMain
16351 JournalNode
```

主要查看是否有 NodeManager 进程；

可通过浏览器访问：<http://192.168.251.18:23188> 查看进程的健康状态（IP 地址替换为上面图中状态为 active 的机器主机名或 IP 地址，如果页面能够成

功显示，则说明启动正常，如果无法打开页面，则说明启动有问题)

4. 登陆 hadoop2 节点上。

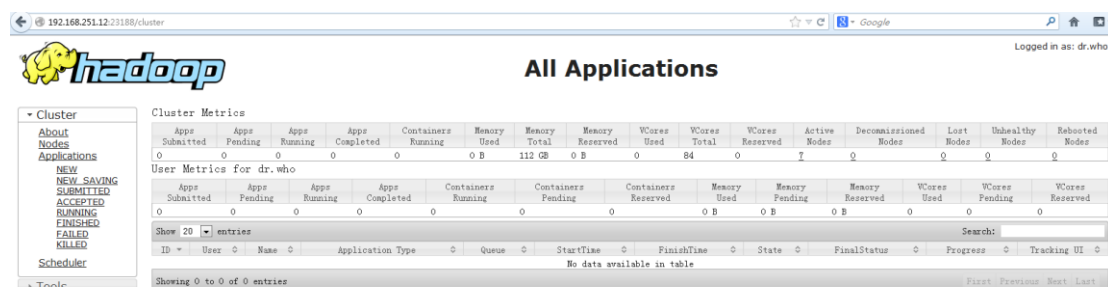
5. 根据 mapred-site.xml 中的 mapreduce.jobhistory.address 配置，可知，historyserver 进程配置在 hadoop2 中，故可在 hadoop2 中启动 historyserver 进程，启动命令：

```
[hadoop@hadoop1 ~]$ mr-jobhistory-daemon.sh start historyserver
```

```
[hadoop@hadoop2 sbin]$ jps
5644 DataNode
5901 NodeManager
3659 Jps
5539 NameNode
5746 DFSZKFailoverController
6142 JobHistoryServer
```

主要查看是否有 JobHistoryServer 进程；

可通过浏览器访问：<http://hadoop2-IP:19888/jobhistory> 查看进程的健康状态（如果能正常显示下面的页面，则表示 yarn 启动成功，hadoop2-IP 替换成 hadoop2 机器对应的 IP 地址）



hadoop 集群的运行日志存放在 /home/hadoop/hadoop/logs 目录下，若安装过程中发现故障可以先分析该目录下的日志文件。

已知问题解决：

- (1) 当 hadoop 启动后，master 一切正常，当其他节点机器有少量 hadoop 未启动的话，需要删除 /data1/dfs 目录下所有内容，然后重启机器，再启动 hadoop，即可恢复正常。注意：请慎重执行此操作，会造成数据的丢失。
- (2) 当 hadoop 启动后，master 一切正常，最后一台启动正常，其他节点机器均不启动，这种情况可能是在配置 slaves 文件时，是在 windows 下修改

的，这可能导致 slaves 文件被存为 dos 格式。

4.12.3.2 停止 Hadoop

请以 **hadoop** 用户身份执行下列命令和配置。

登录 hadoop 集群的 master 机器（在启动了 hbase 的机器上，要先停止 hbase，才可停止 hadoop，具体停止 hbase 服务的方法参考 hbase 相关章节）

任意目录下，执行

```
[hadoop@hadoop2 ~]$ sh /home/hadoop/hadoop/sbin/stop-all.sh
```

命令停止 hadoop

执行上述命令后，用 jps 查看应该看不到 hadoop 相关进程。

4.13 hbase 安装配置

hbase 是基于 hadoop 集群的列式存储数据库，所以必须先安装配置好 hadoop 集群，且正常启动后，方可开始安装配置 hbase。

以下步骤需要在 **hadoop 集群中每一台节点上执行，若有差别，会强调说明。**

4.13.1 安装 hbase 程序

Root 用户下 yum 安装 hbase 软件：

hbase 版本：hbase-0.98.6-cdh5.2.1

Yum 安装：

```
[root@nosqlslave1 ~]$yum clean all
```

```
[root@nosqlslave1 ~]$yum install hbase
```

4.13.2 配置 hbase

请以 **hadoop** 用户身份执行下列命令和配置。

以下文件均在 /home/hadoop/hbase/conf 目录下

4.13.2.1 配置 hbase-env.sh 文件

找到此文件中有以下内容的行（一般是第 35 行）：

```
export HBASE_HEAPSIZE=16384
```

16384 对应的是 32G 内存的服务器。如果服务器内存为 32G，则不需要修改这个值；如果服务器的内存不为 32G，则按比例修改这个值，如：服务器内存为 48G，则这个值应改为：24576。

注意：等号两边一定不能有空格！

4.13.2.2 配置 hbase-site.xml 文件

按照以下说明进行配置（只修改标红的地方，请勿将下面的内容直接粘贴到服务器上的文件中）：

```
<?xml version="1.0"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
<configuration>
  <property>
    <name>hbase.rootdir</name>
    <value>hdfs://runcdhint:8020/hbase</value>
  </property>
  <property>
    <name>hbase.master</name>
    <value>hadoop1:60000</value>
  </property>
  <property>
    <name>hbase.zookeeper.quorum</name>
    <value>hadoop3,hadoop4,hadoop5</value>
  </property>
  <property>
    <name>hbase.tmp.dir</name>
    <value>file:///home/hadoop/hbase/tmp</value>
  </property>
  <property>
    <name>hbase.zookeeper.property.clientPort</name>
    <value>2181</value>
  </property>
  <property>
    <name>hbase.zookeeper.property.dataDir</name>
    <value>/data1/zookeeper/data</value>
  </property>
  <property>
    <name>hbase.cluster.distributed</name>
    <value>true</value>
  </property>
</configuration>
```

hbase.master: hbase master 所在的机器

hbase.zookeeper.quorum: hbase 使用的 zookeeper 所在的机器

hbase.tmp.dir: hbase 存放临时文件的目录，默认可以直接使用上面配置中的路径

hbase.zookeeper.property.dataDir: zookeeper 数据存放的路径，值与 zookeeper 中的 zoo.cfg 文件中的 dataDir=/data1/zookeeper/data 相同

4.13.2.3 配置 regionservers 文件

请清空此文件中的内容，然后将 HBase 集群所有 regionserver 服务器（也就是 Slave 服务器）的主机名称写入到此文件中，一行一个，如下所示：

```
hadoop1
hadoop2
hadoop3
```

4.13.3 hbase 的使用

请以 **hadoop** 用户身份执行下列命令和配置。

4.13.3.1 启动 hbase

(1)启动 hbase 之前需要先启动 zookeeper 和 hadoop, 如果 zookeeper 和 hadoop 已启动, 可省略此操作

执行 4. 11. 3 启动 zookeeper 章节中的命令;

执行 4. 12. 3. 1 启动 hadoop 章节中的命令;

(2) 启动 hbase

登录 hbase 集群的 master 机器

执行 `sh /home/hadoop/hbase/bin/start-hbase.sh` 命令启动 hbase

启动后, 在任意目录下, 用 `jps` 命令, 查看 hbase 启动情况

1、 master 主节点上

```
[hadoop@hadoop1 ~]$ jps
30787 NameNode
30927 DataNode
21363 Jps
31363 ResourceManager
31203 DFSZKFailoverController
418 HMaster
570 HRegionServer
31472 NodeManager
```

主要查看 HMaster 和 HregionServer 进程是否启动;

2、 其他节点上, 机器不同所起进程不同

```
[hadoop@hadoop2 sbin]$ jps
5644 DataNode
5901 NodeManager
5539 NameNode
3798 Jps
5746 DFSZKFailoverController
7431 HRegionServer
6142 JobHistoryServer
```

```
[hadoop@hadoop3 ~]$ jps
16960 DataNode
17157 NodeManager
16620 QuorumPeerMain
17913 HRegionServer
16351 JournalNode
32336 Jps
```

```
[hadoop@hadoop4 ~]$ jps
6874 JournalNode
7605 NodeManager
7198 QuorumPeerMain
8437 HRegionServer
22158 Jps
7445 DataNode
```

```
[hadoop@hadoop5 ~]$ jps
26461 JournalNode
27038 DataNode
```

```
27220 NodeManager
27980 HRegionServer
26762 QuorumPeerMain
9641 Jps
```

主要查看 HregionServer 是否启动;

已知问题解决:

当 hbase 启动后, master 一切正常, 最后一台启动正常, 其他节点机器均不启动, 这种情况可能是在配置 regionserver 文件时, 是在 windows 下修改的, 这可能导致 regionserver 文件被存为 dos 格式。

4.13.3.2 停止 hbase

登录 hbase 集群的 master 机器

任意目录下, 执行 `sh /home/hadoop/hbase/bin/stop-hbase.sh` 命令停止 hbase